EXERCISE
You have a database with the following schema:

Employee (name, address, phone, dept)
Department (name, floor, manager, dept, area, budget)
Contracts (company, contact, area, rating, state)

E.dept is a foreign key to D.dept
C.area is a foreign key to D.area
D.budget is between $5000 and $30000
C.state has 50 unique values
C.rating is between 1 and 10

There are 10,000 employees, 10 per page, 1,000 pages
There are 50 departments, 10 per page, 5 pages
There are 1,000,000 contracts, 100 per page, 10,000 pages

For B-Tree indexes, there are 100 keys per node.

Unclustered B+Tree Index on E.dept, 50 unique values
Unclustered B+Tree Index on D.dept, 50 unique values
Unclustered Hash Index on C.company, 2,000 unique values
Clustered B+Tree Index on C.state, C.rating

Query 1

```sql
SELECT * 
FROM employee E, department D 
WHERE E.dept = D.dept AND D.budget > $10000
```

1) Begin the process of query optimization, by determining all the cheapest and interesting access methods for each relation and their costs. (Pass 1)

2) Enumerate all the two-way joins that the optimizer should estimate costs for, you can use nested loops, index nested loops, hash, and sort-merge join algorithms. You do not need to estimate the costs for each plan

Query 2

```sql
SELECT * 
FROM employee E, department D, contracts C 
WHERE E.dept = D.dept AND D.area = C.area AND 
  D.budget > $10000 AND C.state = CA AND C.rating > 5
```

1) Enumerate and estimate costs for all plans in Pass 1

2) Enumerate plans considered in Pass 2